I 3

```
   CCCCCCCCCCCC  LLL                  IIIIIIIII    UUU         UUU   TTTTTTTTTTTTTTT  LLL
   CCCCCCCCCCCC  LLL                  IIIIIIIII    UUU         UUU   TTTTTTTTTTTTTTT  LLL
   CCCCCCCCCCCC  LLL                  IIIIIIIII    UUU         UUU   TTTTTTTTTTTTTTT  LLL
CCC              LLL                     III       UUU         UUU         TTT        LLL
CCC              LLL                     III       UUU         UUU         TTT        LLL
CCC              LLL                     III       UUU         UUU         TTT        LLL
CCC              LLL                     III       UUU         UUU         TTT        LLL
CCC              LLL                     III       UUU         UUU         TTT        LLL
CCC              LLL                     III       UUU         UUU         TTT        LLL
CCC              LLL                     III       UUU         UUU         TTT        LLL
CCC              LLL                     III       UUU         UUU         TTT        LLL
CCC              LLL                     III       UUU         UUU         TTT        LLL
CCC              LLL                     III       UUU         UUU         TTT        LLL
CCC              LLL                     III       UUU         UUU         TTT        LLL
CCC              LLL                     III       UUU         UUU         TTT        LLL
CCC              LLL                     III       UUU         UUU         TTT        LLL
CCC              LLL                     III       UUU         UUU         TTT        LLL
CCC              LLL                     III       UUU         UUU         TTT        LLL
   CCCCCCCCCCCC  LLLLLLLLLLLLLLL      IIIIIIIII    UUUUUUUUUUUUUUU         TTT        LLLLLLLLLLLLLLL
   CCCCCCCCCCCC  LLLLLLLLLLLLLLL      IIIIIIIII    UUUUUUUUUUUUUUU         TTT        LLLLLLLLLLLLLLL
   CCCCCCCCCCCC  LLLLLLLLLLLLLLL      IIIIIIIII    UUUUUUUUUUUUUUU         TTT        LLLLLLLLLLLLLLL
```

```
**FILE**ID**SETPROCES                                D 3




  SSSSSSSS  EEEEEEEEEE  TTTTTTTTTT  PPPPPPPP    RRRRRRRR      OOOOOO      CCCCCCCC  EEEEEEEEEE    SSSSSSSS
  SSSSSSSS  EEEEEEEEEE  TTTTTTTTTT  PPPPPPPP    RRRRRRRR      OOOOOO      CCCCCCCC  EEEEEEEEEE    SSSSSSSS
SS          EE              TT      PP      PP  RR      RR  OO      OO  CC          EE          SS
SS          EE              TT      PP      PP  RR      RR  OO      OO  CC          EE          SS
SS          EE              TT      PP      PP  RR      RR  OO      OO  CC          EE          SS
SS          EE              TT      PP      PP  RR      RR  OO      OO  CC          EE          SS
  SSSSSS    EEEEEEEE        TT      PPPPPPPP    RRRRRRRR    OO      OO  CC          EEEEEEEE      SSSSSS
  SSSSSS    EEEEEEEE        TT      PPPPPPPP    RRRRRRRR    OO      OO  CC          EEEEEEEE      SSSSSS
        SS  EE              TT      PP          RR  RR      OO      OO  CC          EE                  SS
        SS  EE              TT      PP          RR  RR      OO      OO  CC          EE                  SS
        SS  EE              TT      PP          RR    RR    OO      OO  CC          EE                  SS      ....
        SS  EE              TT      PP          RR    RR    OO      OO  CC          EE                  SS      ....
SSSSSSSS    EEEEEEEEEE      TT      PP          RR      RR    OOOOOO      CCCCCCCC  EEEEEEEEEE  SSSSSSSS        ....
SSSSSSSS    EEEEEEEEEE      TT      PP          RR      RR    OOOOOO      CCCCCCCC  EEEEEEEEEE  SSSSSSSS        ....

LL            IIIIII      SSSSSSSS
LL            IIIIII      SSSSSSSS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II        SSSSSS
LL              II        SSSSSS
LL              II              SS
LL              II              SS
LL              II              SS
LL              II              SS
LLLLLLLLLL    IIIIII    SSSSSSSS
LLLLLLLLLL    IIIIII    SSSSSSSS
```

```
0 MODULE setproces ( IDENT = 'V04-000',
0                  ADDRESSING_MODE (EXTERNAL = GENERAL)) =
1 BEGIN
1
1 !****************************************************************************
1 !*                                                                          *
1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                  *
1 !*  ALL RIGHTS RESERVED.                                                    *
1 !*                                                                          *
1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
1 !*  TRANSFERRED.                                                            *
1 !*                                                                          *
1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
1 !*  CORPORATION.                                                            *
1 !*                                                                          *
1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
1 !*                                                                          *
1 !*                                                                          *
1 !****************************************************************************
1
1 !++
1 ! FACILITY:  SETPRO Command
1 !
1 ! ABSTRACT:
1 !
1 !       This module sets various parameters for a process.
1 !
1 ! ENVIRONMENT:
1 !
1 !       VAX/VMS operating system, user mode
1 !
1 ! AUTHOR:  Gerry Smith                            12-Jan-1983
1 !
1 ! Modified by:
1 !
1 !       V03-007 AEW0002         Anne Warner             05-Jul-1984
1 !               Change ALTPRV to ALTPRI from previous fix because
1 !               there's no such creature as ALPTRV.
1 !
1 !       V03-006 AEW0001         Anne Warner             04-Jun-1984
1 !               Add non-fatal error message SET$_NOPRIO indicating
1 !               that the process priority could not be raised above
1 !               base priority because it does not have the user
1 !               privilege ALTPRV. (actually its ALTPRI - see  above)
1 !
1 !       V03-005 GAS0182         Gerry Smith             19-Sep-1983
1 !               Change the way that privileges get set.  Instead of
1 !               disabling all privileges and then re-enabling them,
1 !               figure out which privileges to enable, and which
1 !               to disable, and then do it explicitly.
```

```
1 !
1 !      V03-004 GAS0157          Gerry Smith             25-Jul-1983
1 !              Use the real process ID of the process, rather than
1 !              what the user input.
1 !
1 !      V03-003 WMC0001          Wayne Cardoza           11-Apr-1983
1 !              Add SET PROC/DUMP.
1 !
1 !      V03-002 GAS0113                   Gerry Smith    30-Mar-1983
1 !              Collect and validate all qualifiers and values first,
1 !              then make all the modifications.
1 !
1 !      V03-001 GAS0112                   Gerry Smith    29-Mar-1983
1 !              Remove last traces of the old command dispatcher.
1 !
1 !--
```

```
:     75          0074  1 !
:     76          0075  1 ! Include files
:     77          0076  1 !
:     78          0077  1 LIBRARY 'SYS$LIBRARY:LIB';              ! VAX/VMS common definitions
:     79          0078  1
:     80          0079  1
```

```
   1 !
   1 ! Table of contents
   1 !
   1
   1 FORWARD ROUTINE
   1     set$process : NOVALUE,                ! Main routine
   1     get_name : NOVALUE,                   ! Get process name
   1     get_quals : NOVALUE,                  ! Get all qualifiers and values
   1     set_process  : NOVALUE,               ! Set them
   1     set_dump     : NOVALUE;               ! Kernel mode routine to set dump flag
   1
   1 !
   1 ! External routines
   1 !
   1 EXTERNAL ROUTINE
   1     lib$cvt_htb,                          ! Convert ASCII (hex) to binary
   1     lib$cvt_dtb,                          ! Convert ASCII (decimal) to binary
   1     prv$setpriv,                          ! Set/clear privilege bits in bitmask
   1     cli$get_value,                        ! Get value from CLI
   1     cli$present;                          ! See if qualifier is present
   1
   1 !
   1 ! External globals
   1 !
   1 EXTERNAL
   1     ctl$gq_procpriv : VECTOR[2],          ! Process privileges
   1     ctl$gl_phd  : REF BLOCK[,BYTE];       ! P1 window to PHD
   1
   1 !
   1 ! Declare the final status return.
   1 !
   1 EXTERNAL
   1     set$exit_status;
   1
   1
   1 !
   1 ! Declare some shared messages
   1 !
P  1 $SHR_MSGDEF     (SET,119,LOCAL,
   1                 (invquaval,     error));
   1
   1 !
   1 ! Declare literals defined elsewhere
   1 !
   1 EXTERNAL LITERAL
   1     cli$_absent,                          ! Qualifier absent
   1     cli$_negated,                         ! Qualifier explicitly negated
   1     set$_writeerr,                        ! Error modifying
   1     set$_noprio,                          ! Priority not changed
   1     set$_prioset,                         ! Priority changed
   1     set$_noname,                          ! Name not changed,
   1     set$_nameset,                         ! Name changed
   1     set$_notsuspnd,                       ! Process not suspended
   1     set$_resumed,                         ! Process resumed
   1     set$_notresumed,                      ! Process not resumed
   1     set$_suspnd,                          ! Process suspended
   1     set$_modeset,                         ! Process mode changed
```

```
;   139          0137  1        set$_notpriv,                   ! Privileges not set
;   140          0138  1        set$_privset,                   ! Privileges set
;   141          0139  1        set$_ownproc;                   ! Qualifier only good for own process
;   142          0140  1
;   143          0141  1  !
;   144          0142  1  ! Declare the literals for the different qualifiers
;   145          0143  1  !
;   146          0144  1  LITERAL
;   147        P 0145  1        $EQULST (set$_,,1,1,
;   148        P 0146  1            (log,),
;   149        P 0147  1            (priority,),
;   150        P 0148  1            (name,),
;   151        P 0149  1            (resume,),
;   152        P 0150  1            (suspend,),
;   153        P 0151  1            (swap,),
;   154        P 0152  1            (swapval,),
;   155        P 0153  1            (wait,),
;   156        P 0154  1            (waitval),
;   157        P 0155  1            (priv),
;   158        P 0156  1            (dump),
;   159          0157  1            (dumpval));
;   160          0158  1
;   161          0159  1  !
;   162          0160  1  ! It is convenient to declare one large vector containing all the data,
;   163          0161  1  ! and give the separate pieces names that humans like.  So, declare a
;   164          0162  1  ! macro that will make those binds at the beginning of each subroutine.
;   165          0163  1  !
;   166          0164  1  MACRO
;   167        M 0165  1      BIND_DATA =
;   168        M 0166  1          BIND
;   169        M 0167  1              flags       = data_buffer[0] : BITVECTOR[32],
;   170        M 0168  1              pid         = data_buffer[1] : VOLATILE,
;   171        M 0169  1              priority    = data_buffer[2],
;   172        M 0170  1              new_name    = data_buffer[3] : VECTOR[2],
;   173        M 0171  1              enab_priv   = data_buffer[5] : VECTOR[2],
;   174        M 0172  1              disab_priv  = data_buffer[7] : VECTOR[2],
;   175        M 0173  1              name_desc   = data_buffer[9] : VECTOR[2],
;   176          0174  1              name_buffer = data_buffer[11]: VECTOR[3];%;
;  178          0175  1 GLOBAL ROUTINE set$process : NOVALUE =
;  179          0176  2 BEGIN
;  180          0177  2 !++
;  181          0178  2 ! Functional description
;  182          0179  2 !
;  183          0180  2 !       This is the routine for the SET PROCESS command.  It is called from the
;  184          0181  2 !       SET command processor,  and sets various runtime parameters for a
;  185          0182  2 !       process.
;  186          0183  2 !
;  187          0184  2 ! Inputs
;  188          0185  2 !       None
;  189          0186  2 !
;  190          0187  2 ! Outputs
;  191          0188  2 !       None
;  192          0189  2 !
;  193          0190  2 !----
;  194          0191  2
;  195          0192  2 LOCAL
;  196          0193  2     status,                           ! Status return
;  197          0194  2     data_buffer : VECTOR[20]          ! Buffer containing all the data
;  198          0195  2             INITIAL(REP 20 of (0));   ! initially clear
;  199          0196  2
;  200          0197  2 get_name(data_buffer);                ! Get the name of the process.
;  201          0198  2
;  202          0199  2 get_quals(data_buffer);               ! Get all the qualifiers.
;  203          0200  2
;  204          0201  2 IF .set$exit_status                   ! If no errors so far,
;  205          0202  2 THEN set_process(data_buffer);        ! set the new values.
;  206          0203  2
;  207          0204  2 RETURN;
;  208          0205  1 END;


                                                   .TITLE  SETPROCES
                                                   .IDENT  \V04-000\

                                                   .PSECT  $PLIT$,NOWRT,NOEXE,2

                         00000000# 00000 P.AAA:    .LONG   0[20]                                        ;

                                                   .EXTRN  LIB$CVT_HTB, LIB$CVT_DTB
                                                   .EXTRN  PRV$SETPRIV, CLI$GET_VALUE
                                                   .EXTRN  CLI$PRESENT, CTL$GQ_PROCPRIV
                                                   .EXTRN  CTL$GL_PHD, SET$EXIT_STATUS
                                                   .EXTRN  CLI$_ABSENT, CLI$_NEGATED
                                                   .EXTRN  SET$_WRITEERR, SET$_NOPRIO
                                                   .EXTRN  SET$_PRIOSET, SET$_NONAME
                                                   .EXTRN  SET$_NAMESET, SET$_NOTSUSPND
                                                   .EXTRN  SET$_RESUMED, SET$_NOTRESUMED
                                                   .EXTRN  SET$_SUSPND, SET$_MODESET
                                                   .EXTRN  SET$_NOTPRIV, SET$_PRIVSET
                                                   .EXTRN  SET$_OWNPROC

                                                   .PSECT  $CODE$,NOWRT,2

                                 003C 00000        .ENTRY  SET$PROCESS, Save R2,R3,R4,R5        ; 0175
                 5E       B0   AE   9E 00002        MOVAB   -80(SP), SP                          ;
```

```
                6E     0000'  CF       0050    8F   28 00006          MOVC3   #80, P.AAA, DATA_BUFFER          ; 0195
                                               5E   DD 0000E          PUSHL   SP                               ; 0197
                       0000V  CF               01   FB 00010          CALLS   #1, GET_NAME                     ;
                                               5E   DD 00015          PUSHL   SP                               ; 0199
                       0000V  CF               01   FB 00017          CALLS   #1, GET_QUALS                    ;
                              07 00000000G     00   E9 0001C          BLBC    SET$EXIT_STATUS, 1$              ; 0201
                                               5E   DD 00023          PUSHL   SP                               ; 0202
                       0000V  CF               01   FB 00025          CALLS   #1, SET_PROCESS                  ;
                                                    04 0002A 1$:      RET                                      ; 0205

; Routine Size:  43 bytes,    Routine Base:  $CODE$ + 0000
```

```
  1 ROUTINE get_name (data_buffer) : NOVALUE =
  2 BEGIN
  2
  2 !++
  2 !
  2 ! Get the process name and tuck it away to use later.
  2 !
  2 ! Inputs
  2 !       DATA_BUFFER - contains all the data cells
  2 !
  2 ! Outputs
  2 !       NAME_DESC will point to the process name
  2 !       PID will contain the process ID of the process to change
  2 !
  2 !--
  2
  2 MAP
  2     data_buffer : REF VECTOR;
  2
  2 LOCAL
  2     status,                             ! General status return
  2     desc : $BBLOCK[dsc$c_s_bln],        ! General descriptor
  2     iosb : VECTOR[4,WORD],              ! Status block for GETJPI
  2     jpi_list : $ITMLST_DECL(ITEMS = 2); ! Item list for GETJPI
  2
  2 !
  2 ! Bind the data to names we can understand
  2 !
  2 bind_data;
  2
  2 !
  2 ! Collect the process name, if specified.  If no process name is
  2 ! specified, try a process id.
  2 !
  2 $init_dyndesc(desc);                         ! Make the descriptor dynamic
  2 pid = 0;                                     ! Show that no PID found yet.
  2 name_desc[1] = name_buffer;                  ! Point to process name buffer
  2
  2 !
  2 ! If the process name is given, also get the PID
  2 !
  2 IF cli$get_value(%ASCID 'PROCESS', desc)! Get the process name
  2 THEN                                         ! If the process name exists,
  3     BEGIN                                    ! convert it to a PID.
  3 !
  3 ! Set up the JPI item list to get the PID.
  3 !
P 3     $ITMLST_INIT(ITMLST = jpi_list,
  3                 (ITMCOD = jpi$_pid, BUFADR = pid));
P 3     status = $GETJPIW(ITMLST = jpi_list,
P 3                       PRCNAM = desc,
  3                       IOSB   = iosb);
  3     IF .status
  3     THEN status = .iosb[0];
  3     IF NOT .status
  3     THEN SIGNAL(set$_writeerr, 1, desc, .status)
  3     ELSE
  4         BEGIN
  4         CH$MOVE(.desc[dsc$w_length], .desc[dsc$a_pointer], name_buffer);
  4         name_desc[0] = .desc[dsc$w_length];
  4         name_desc[1] = name_buffer;
  3         END;
  3     END
  3   !
  3   ! If no process name, perhaps the PID was specified.
  3   !
  2 ELSE                                     ! If no process name,
  3     BEGIN                                ! try for a PID
  3     IF cli$get_value(%ASCID 'IDENTIFICATION',
  3                      desc)
  3     THEN                                 ! If we get a PID,
  4         BEGIN                            ! convert it to a number
  5         IF NOT (status = lib$cvt_htb(.desc[dsc$w_length],
  5                                      .desc[dsc$a_pointer],
  5                                      pid))
  4         THEN SIGNAL(set$_invquaval, 2,
  4                     desc, %ASCID 'IDENTIFICATION')
  4         ELSE
  5             BEGIN
P 5             $ITMLST_INIT(ITMLST = jpi_list,
P 5                         (ITMCOD = jpi$_pid,
P 5                          BUFADR = pid),
P 5                         (ITMCOD = jpi$_prcnam,
P 5                          BUFADR = name_buffer,
P 5                          BUFSIZ = 20,
  5                          RETLEN = name_desc[0]));
P 5             status = $GETJPIW(ITMLST = jpi_list,
P 5                               PIDADR = pid,
  5                               IOSB   = iosb);
  5             IF .status
  5             THEN status = .iosb[0];
  5             IF NOT .status
  5             THEN SIGNAL(set$_writeerr, 1, desc, .status);
  4             END;
  3         END;
  2     END;
  2   !
  2   ! If no PID specified, use the PID and name of the current process.
  2   !
  2 IF .pid EQL 0
  2 THEN
  3     BEGIN
P 3     $ITMLST_INIT(ITMLST = jpi_list,             ! Set up JPI list to get
P 3                 (ITMCOD = jpi$_pid,             ! the current process
P 3                  BUFADR = pid),                 ! PID and name, and
P 3                 (ITMCOD = jpi$_prcnam,          ! stuff them into the
P 3                  BUFADR = name_buffer,          ! appropriate places.
P 3                  BUFSIZ = 20,
  3                  RETLEN = name_desc[0]));
P 3     status = $GETJPIW(ITMLST = jpi_list,
  3                       IOSB   = iosb);
  3     IF .status
  3     THEN status = .iosb[0];
```

```
;  324          0320  3         IF NOT .status
;  325          0321  3         THEN SIGNAL(set$_writeerr, 1, %ASCID 'this process', .status);
;  326          0322  2         END;
;  327          0323  2
;  328          0324  2 RETURN;
;  329          0325  1 END;



                                                                    .PSECT  $PLIT$,NOWRT,NOEXE,2

                       00  53  53  45  43  4F  52  50   00050 P.AAC:  .ASCII  \PROCESS\<0>                          :
                                            010E0007    00058 P.AAB:  .LONG   17694727                              :
                                           00000000'    0005C         .ADDRESS P.AAC                                :
00  4E  4F  49  54  41  43  49  46  49  54  4E  45  44  49   00060 P.AAE:  .ASCII  \IDENTIFICATION\<0><0>          :
                                                  00    0006F                                                       :
                                            010E000E    00070 P.AAD:  .LONG   17694734                              :
                                           00000000'    00074         .ADDRESS P.AAE                                :
00  4E  4F  49  54  41  43  49  46  49  54  4E  45  44  49   00078 P.AAG:  .ASCII  \IDENTIFICATION\<0><0>          :
                                                  00    00087                                                       :
                                            010E000E    00088 P.AAF:  .LONG   17694734                              :
                                           00000000'    0008C         .ADDRESS P.AAG                                :
            73  73  65  63  6F  72  70  20  73  69  68  74   00090 P.AAI:  .ASCII  \this process\                :
                                            010E000C    0009C P.AAH:  .LONG   17694732                              :
                                           00000000'    000A0         .ADDRESS P.AAI                                :

                                                                    .EXTRN  SYS$GETJPIW

                                                                    .PSECT  $CODE$,NOWRT,2

                                           OFFC 00000 GET_NAME:
                                                                    .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11      : 0206
                        5B 00000000G  00  9E 00002              MOVAB   CLI$GET_VALUE, R11                             :
                        5A 00000000G  00  9E 00009              MOVAB   SYS$GETJPIW, R10                               :
                        5E            2C  C2 00010              SUBL2   #44, SP                                        :
                        50       04   AC  D0 00013              MOVL    DATA_BUFFER, R0                                : 0229
                        58       04   A0  9E 00017              MOVAB   4(R0), R8                                      :
                        57       24   A0  9E 0001B              MOVAB   36(R0), R7                                     :
                        59       2C   A0  9E 0001F              MOVAB   44(R0), R9                                     :
               24       AE 020E0000   8F  D0 00023              MOVL    #34471936, DESC                                : 0240
                                 28   AE  D4 0002B              CLRL    DESC+4                                         :
                                      68  D4 0002E              CLRL    (R8)                                           : 0241
               04       A7            59  D0 00030              MOVL    R9, 4(R7)                                      : 0242
                                 24   AE  9F 00034              PUSHAB  DESC                                           : 0247
                              0000'   CF  9F 00037              PUSHAB  P.AAB                                          :
                        6B            02  FB 0003B              CALLS   #2, CLI$GET_VALUE                              :
                        3D            50  E9 0003E              BLBC    R0, 1$                                         :
                        50            6E  9E 00041              MOVAB   JPI_LIST, $$ITMBLKPTR                          : 0254
                        80 03190004   8F  D0 00044              MOVL    #51970052, ($$ITMBLKPTR)+                      :
                        80            58  D0 0004B              MOVL    R8, ($$ITMBLKPTR)+                             :
                                      80  7C 0004E              CLRQ    ($$ITMBLKPTR)+                                 :
                                      7E  7C 00050              CLRQ    -(SP)                                          : 0257
                                 24   AE  9F 00052              PUSHAB  IOSB                                           :
                                 0C   AE  9F 00055              PUSHAB  JPI_LIST                                       :
                                 34   AE  9F 00058              PUSHAB  DESC                                           :
                                      7E  7C 0005B              CLRQ    -(SP)                                          :
                        6A            07  FB 0005D              CALLS   #7, SYS$GETJPIW                                :
```

```
                        56            50  D0 00060       MOVL     R0, STATUS                              ;
                        7E            56  E9 00063       BLBC     STATUS, 3$                              ; 0258
                        56        1C  AE  3C 00066       MOVZWL   IOSB, STATUS                            ; 0259
                        77            56  E9 0006A       BLBC     STATUS, 3$                              ; 0260
        69        28    BE        24  AE  28 0006D       MOVC3    DESC, @DESC+4, (R9)                     ; 0264
                        67        24  AE  3C 00073       MOVZWL   DESC, (R7)                              ; 0265
                  04    A7            59  D0 00077       MOVL     R9, 4(R7)                               ; 0266
                                    0084  31 0007B       BRW      6$                                      ; 0247
                                  24  AE  9F 0007E 1$:   PUSHAB   DESC                                    ; 0275
                               0000'  CF  9F 00081       PUSHAB   P.AAD                                   ;
                        6B            02  FB 00085       CALLS    #2, CLI$GET_VALUE                       ;
                        77            50  E9 00088       BLBC     R0, 6$                                  ;
                                      58  DD 0008B       PUSHL    R8                                      ; 0279
                                  2C  AE  DD 0008D       PUSHL    DESC+4                                  ; 0280
                        7E        2C  AE  3C 00090       MOVZWL   DESC, -(SP)                             ; 0279
             00000000G  00            03  FB 00094       CALLS    #3, LIB$CVT_HTB                         ;
                        56            50  D0 0009B       MOVL     R0, STATUS                              ;
                        11            56  E8 0009E       BLBS     STATUS, 2$                              ;
                               0000'  CF  9F 000A1       PUSHAB   P.AAF                                   ; 0283
                                  28  AE  9F 000A5       PUSHAB   DESC                                    ; 0282
                                      02  DD 000A8       PUSHL    #2                                      ;
                           0077132A   8F  DD 000AA       PUSHL    #7803690                                ;
                                      49  11 000B0       BRB      5$                                      ;
                        50            6E  9E 000B2 2$:   MOVAB    JPI_LIST, $$ITMBLKPTR                   ; 0292
                        80  03190004  8F  D0 000B5       MOVL     #51970052, ($$ITMBLKPTR)+               ;
                        80            58  D0 000BC       MOVL     R8, ($$ITMBLKPTR)+                      ;
                                      80  D4 000BF       CLRL     ($$ITMBLKPTR)+                          ;
                        80  031C0014  8F  D0 000C1       MOVL     #52166676, ($$ITMBLKPTR)+               ;
                        80            59  D0 000C8       MOVL     R9, ($$ITMBLKPTR)+                      ;
                        80            57  D0 000CB       MOVL     R7, ($$ITMBLKPTR)+                      ;
                                      80  D4 000CE       CLRL     ($$ITMBLKPTR)+                          ;
                                      7E  7C 000D0       CLRQ     -(SP)                                   ; 0295
                                  24  AE  9F 000D2       PUSHAB   IOSB                                    ;
                                  0C  AE  9F 000D5       PUSHAB   JPI_LIST                                ;
                                      7E  D4 000D8       CLRL     -(SP)                                   ;
                                      58  DD 000DA       PUSHL    R8                                      ;
                                      7E  D4 000DC       CLRL     -(SP)                                   ;
                        6A            07  FB 000DE       CALLS    #7, SYS$GETJPIW                         ;
                        56            50  D0 000E1       MOVL     R0, STATUS                              ;
                        07            56  E9 000E4 3$:   BLBC     STATUS, 4$                              ; 0296
                        56        1C  AE  3C 000E7       MOVZWL   IOSB, STATUS                            ; 0297
                        14            56  E8 000EB       BLBS     STATUS, 6$                              ; 0298
                                      56  DD 000EE 4$:   PUSHL    STATUS                                  ; 0299
                                  28  AE  9F 000F0       PUSHAB   DESC                                    ;
                                      01  DD 000F3       PUSHL    #1                                      ;
                           00000000G  8F  DD 000F5       PUSHL    #SET$_WRITEERR                          ;
             00000000G  00            04  FB 000FB 5$:   CALLS    #4, LIB$SIGNAL                          ;
                                      68  D5 00102 6$:   TSTL     (R8)                                    ; 0306
                                      4F  12 00104       BNEQ     8$                                      ;
                        50            6E  9E 00106       MOVAB    JPI_LIST, $$ITMBLKPTR                   ; 0315
                        80  03190004  8F  D0 00109       MOVL     #51970052, ($$ITMBLKPTR)+               ;
                        80            58  D0 00110       MOVL     R8, ($$ITMBLKPTR)+                      ;
                                      80  D4 00113       CLRL     ($$ITMBLKPTR)+                          ;
                        80  031C0014  8F  D0 00115       MOVL     #52166676, ($$ITMBLKPTR)+               ;
                        80            59  D0 0011C       MOVL     R9, ($$ITMBLKPTR)+                      ;
                        80            57  D0 0011F       MOVL     R7, ($$ITMBLKPTR)+                      ;
                                      80  D4 00122       CLRL     ($$ITMBLKPTR)+                          ;
```

```
                                   7E  7C 00124       CLRQ    -(SP)                                  ; 0317
                        24         AE  9F 00126       PUSHAB  IOSB                                   ;
                        0C         AE  9F 00129       PUSHAB  JPI_LIST                               ;
                                   7E  7C 0012C       CLRQ    -(SP)                                  ;
                                   7E  D4 0012E       CLRL    -(SP)                                  ;
                  6A               07  FB 00130       CALLS   #7, SYS$GETJPIW                        ;
                  56               50  D0 00133       MOVL    R0, STATUS                             ;
                  07               56  E9 00136       BLBC    STATUS, 7$                             ; 0318
                  56      1C       AE  3C 00139       MOVZWL  IOSB, STATUS                           ; 0319
                  15               56  E8 0013D       BLBS    STATUS, 8$                             ; 0320
                                   56  DD 00140 7$:   PUSHL   STATUS                                 ; 0321
                        0000'      CF  9F 00142       PUSHAB  P.AAH                                  ;
                                   01  DD 00146       PUSHL   #1                                     ;
                     00000000G     8F  DD 00148       PUSHL   #SET$_WRITEERR                         ;
       00000000G  00               04  FB 0014E       CALLS   #4, LIB$SIGNAL                         ;
                                       04 00155 8$:   RET                                            ; 0325

; Routine Size:  342 bytes,    Routine Base:  $CODE$ + 002B
```

```
ROUTINE get_quals (data_buffer) : NOVALUE =
BEGIN
!++
!
! Get all and validate all the qualifiers.  If any errors, signal them.
!
! Inputs
!       DATA_BUFFER contains all the data cells.
!
! Outputs
!       FLAGS will have bits set to indicate what is to change.
!       PRIORITY will have the new priority.
!       NEW_NAME will point to the new process name.
!       PRIV will be the new privilege mask.
!


!
! Bind the data buffer to names that humans like.
!

MAP
    data_buffer : REF VECTOR;

LOCAL
    status,
    ourpid,
    iosb : VECTOR[2],
    jpi_list : $ITMLST_DECL(ITEMS = 1),
    desc : $BBLOCK[dsc$c_s_bln];

!
! Bind the data buffer to names that are more understandable
!
bind_data;

!
! Obtain the process ID of this process.  It will be used to check that
! certain qualifiers are not requested inappropriately.
!
$ITMLST_INIT(ITMLST = jpi_list,
            (ITMCOD = jpi$_pid,
             BUFADR = ourpid)
            );
$GETJPIW(ITMLST = jpi_list,
         IOSB   = iosb);


!
! See if logging is requested.
!
flags[set$_log] = cli$present(%ASCID 'LOG');

!
! /PRIORITY=n
!
$init_dyndesc(desc);                                    ! Make desc. dynamic
IF cli$get_value(%ASCID 'PRIORITY', desc)               ! See if qualifier there
```

```
2 THEN
3     BEGIN
3     flags[set$_priority] = 1;
3     IF NOT lib$cvt_dtb(.desc[dsc$w_length],            ! If not a good value,
3                        .desc[dsc$a_pointer],           ! tell the user
3                        priority)
3     THEN SIGNAL(set$_invquaval, 2, desc, %ASCID 'PRIORITY')
3     ELSE
4         BEGIN                                          ! Perform bounds
4         IF .priority GTR 31                            ! checking, telling
4         OR .priority LSS 0                             ! if out of bounds
4         THEN SIGNAL(set$_invquaval, 2, desc, %ASCID 'PRIORITY');
3         END;
2     END;
2
2 !
2 ! /NAME = string
2 !
2 IF cli$get_value(%ASCID 'NAME', desc)                 ! If a new name requested
2 THEN
3     BEGIN
3     IF .ourpid NEQ .pid
3     THEN SIGNAL(set$_ownproc,
3                 1,
3                 %ASCID 'NAME');
3     flags[set$_name] = 1;                              ! Set the flag
3     new_name[0] = .desc[dsc$w_length];                 ! Point to the name
3     new_name[1] = .desc[dsc$a_pointer];
3     $init_dyndesc(desc);                               ! Re-use the descriptor
2     END;
2
2 !
2 ! /SUSPEND and /RESUME are inverses of each other, and so are treated
2 ! together.  However, although there is a /NOSUSPEND, there is no /NORESUME.
2 !
2 status = cli$present(%ASCID 'SUSPEND');
2 IF .status
2 THEN flags[set$_suspend] = 1
2 ELSE IF .status EQL cli$_negated
2 THEN flags[set$_resume] =1;
2 IF cli$present(%ASCID 'RESUME')
2 THEN flags[set$_resume] = 1;
2
2 !
2 ! /[NO]SWAP
2 !
2 status = cli$present(%ASCID 'SWAPPING');
2 IF .status NEQ cli$_absent
2 THEN
3     BEGIN
3     IF .ourpid NEQ .pid
3     THEN SIGNAL(set$_ownproc,
3                 1,
3                 %ASCID '[NO]SWAP');
3     flags[set$_swap] = 1;
3     flags[set$_swapval] = NOT .status;
2     END;
```

```
2
2 !
2 ! /[NO]RESOURCE_WAIT
2 !
2 status = cli$present(%ASCID 'RESOURCE_WAIT');
2 IF .status NEQ cli$_absent
2 THEN
3     BEGIN
3     IF .ourpid NEQ .pid
3     THEN SIGNAL(set$_ownproc,
3                 1,
3                 %ASCID '[NO]RESOURCE_WAIT');
3     flags[set$_wait] = 1;
3     flags[set$_waitval] = NOT .status;
2     END;
2
2 !
2 ! /PRIVILEGES = list
2 !
2 IF cli$present(%ASCID 'PRIVILEGES')
2 THEN
3     BEGIN
3     LOCAL
3         oldpriv : VECTOR[2],
3         newpriv : VECTOR[2];
3     IF .ourpid NEQ .pid
3     THEN SIGNAL(set$_ownproc,
3                 1,
3                 %ASCID 'PRIVILEGES');
3     flags[set$_priv] = 1;
3
3 !
3 ! Copy the current process privileges into local memory.
3 !
3     oldpriv[0] = newpriv[0] = .ctl$gq_procpriv[0];
3     oldpriv[1] = newpriv[1] = .ctl$gq_procpriv[1];
3
3 !
3 ! Then get all the privileges that were specified by the user.  For
3 ! each privilege given, call the unsupported, undocumented routine
3 ! PRV$SETPRIV, which will decipher the ASCII text given it (e.g. NOLOG)
3 ! and set or clear the corresponding bit in the two-longword privilege
3 ! bitmask.
3 !
3     WHILE cli$get_value(%ASCID 'PRIVILEGES', desc)
3     DO
4         BEGIN
5         IF NOT (status = PRV$SETPRIV(desc, newpriv))
4         THEN SIGNAL(set$_invquaval, 2,                 ! Say it's invalid
4                     desc,
4                     %ASCID 'PRIVILEGES');
3         END;
3
3 !
3 ! Get the privileges to enable and disable.
3 !
3     enab_priv[0] = .newpriv[0] AND NOT .oldpriv[0];
```

```
;  502        0497  3      enab_priv[1] = .newpriv[1] AND NOT .oldpriv[1];
;  503        0498  3      disab_priv[0] = .oldpriv[0] AND NOT .newpriv[0];
;  504        0499  3      disab_priv[1] = .oldpriv[1] AND NOT .newpriv[1];
;  505        0500  2      END;
;  506        0501  2
;  507        0502  2   !
;  508        0503  2   ! /[NO]DUMP
;  509        0504  2   !
;  510        0505  2   status = cli$present(%ASCID 'DUMP');
;  511        0506  2   IF .status NEQ cli$_absent
;  512        0507  2   THEN
;  513        0508  3      BEGIN
;  514        0509  3      IF .ourpid NEQ .pid
;  515        0510  3      THEN SIGNAL(set$_ownproc,
;  516        0511  3                 1,
;  517        0512  3                 %ASCID 'DUMP');
;  518        0513  3      flags[set$_dump] = 1;
;  519        0514  3      flags[set$_dumpval] = .status;
;  520        0515  2      END;
;  521        0516  2
;  522        0517  2   RETURN;
;  523        0518  1   END;
; INFO#250              L1:0404
; Referenced LOCAL symbol OURPID is probably not initialized


                                                                     .PSECT  $PLIT$,NOWRT,NOEXE,2

                                   00  47  4F  4C  000A4 P.AAK:   .ASCII   \LOG\<0>
                                       010E0003   000A8 P.AAJ:   .LONG    17694723
                                       00000000'  000AC          .ADDRESS P.AAK
               59  54  49  52  4F  49  52  50  000B0 P.AAM:   .ASCII   \PRIORITY\
                                       010E0008   000B8 P.AAL:   .LONG    17694728
                                       00000000'  000BC          .ADDRESS P.AAM
               59  54  49  52  4F  49  52  50  000C0 P.AAO:   .ASCII   \PRIORITY\
                                       010E0008   000C8 P.AAN:   .LONG    17694728
                                       00000000'  000CC          .ADDRESS P.AAO
               59  54  49  52  4F  49  52  50  000D0 P.AAQ:   .ASCII   \PRIORITY\
                                       010E0008   000D8 P.AAP:   .LONG    17694728
                                       00000000'  000DC          .ADDRESS P.AAQ
                                   45  4D  41  4E  000E0 P.AAS:   .ASCII   \NAME\
                                       010E0004   000E4 P.AAR:   .LONG    17694724
                                       00000000'  000E8          .ADDRESS P.AAS
                                   45  4D  41  4E  000EC P.AAU:   .ASCII   \NAME\
                                       010E0004   000F0 P.AAT:   .LONG    17694724
                                       00000000'  000F4          .ADDRESS P.AAU
               00  44  4E  45  50  53  55  53  000F8 P.AAW:   .ASCII   \SUSPEND\<0>
                                       010E0007   00100 P.AAV:   .LONG    17694727
                                       00000000'  00104          .ADDRESS P.AAW
               00  00  45  4D  55  53  45  52  00108 P.AAY:   .ASCII   \RESUME\<0><0>
                                       010E0006   00110 P.AAX:   .LONG    17694726
                                       00000000'  00114          .ADDRESS P.AAY
               47  4E  49  50  50  41  57  53  00118 P.ABA:   .ASCII   \SWAPPING\
                                       010E0008   00120 P.AAZ:   .LONG    17694728
                                       00000000'  00124          .ADDRESS P.ABA
               50  41  57  53  5D  4F  4E  5B  00128 P.ABC:   .ASCII   \[NO]SWAP\
                                                    010E0008  00130 P.ABB:  .LONG    17694728
                                                    00000000' 00134         .ADDRESS P.ABC
00 00 54 49 41 57 5F 45 43 52 55 4F 53  45  52      00138 P.ABE:  .ASCII   \RESOURCE_WAIT\<0><0><0>
                                                00      00147
                                                    010E000D  00148 P.ABD:  .LONG    17694733
                                                    00000000' 0014C         .ADDRESS P.ABE
41 57 5F 45 43 52 55 4F 53 45 52 5D 4F  4E  5B      00150 P.ABG:  .ASCII   \[NO]RESOURCE_WAIT\<0><0><0>
                              00 00 00  54  49      0015F
                                                    010E0011  00164 P.ABF:  .LONG    17694737
                                                    00000000' 00168         .ADDRESS P.ABG
         00 00 53 45 47 45 4C 49 56 49  52  50      0016C P.ABI:  .ASCII   \PRIVILEGES\<0><0>
                                                    010E000A  00178 P.ABH:  .LONG    17694730
                                                    00000000' 0017C         .ADDRESS P.ABI
         00 00 53 45 47 45 4C 49 56 49  52  50      00180 P.ABK:  .ASCII   \PRIVILEGES\<0><0>
                                                    010E000A  0018C P.ABJ:  .LONG    17694730
                                                    00000000' 00190         .ADDRESS P.ABK
         00 00 53 45 47 45 4C 49 56 49  52  50      00194 P.ABM:  .ASCII   \PRIVILEGES\<0><0>
                                                    010E000A  001A0 P.ABL:  .LONG    17694730
                                                    00000000' 001A4         .ADDRESS P.ABM
         00 00 53 45 47 45 4C 49 56 49  52  50      001A8 P.ABO:  .ASCII   \PRIVILEGES\<0><0>
                                                    010E000A  001B4 P.ABN:  .LONG    17694730
                                                    00000000' 001B8         .ADDRESS P.ABO
                                    50 4D  55  44   001BC P.ABQ:  .ASCII   \DUMP\
                                                    010E0004  001C0 P.ABP:  .LONG    17694724
                                                    00000000' 001C4         .ADDRESS P.ABQ
                                    50 4D  55  44   001C8 P.ABS:  .ASCII   \DUMP\
                                                    010E0004  001CC P.ABR:  .LONG    17694724
                                                    00000000' 001D0         .ADDRESS P.ABS


                                                                            .PSECT  $CODE$,NOWRT,2

                                                 0FFC 00000 GET_QUALS:
                                                                            .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11          ; 0326
                          5B 00000000G    00  9E 00002              MOVAB   CLI$GET_VALUE, R11                             ;
                          5A 00000000G    8F  D0 00009              MOVL    #SET$_OWNPROC, R10                             ;
                          59 00000000G    00  9E 00010              MOVAB   LIB$SIGNAL, R9                                 ;
                          58 00000000G    00  9E 00017              MOVAB   CLI$PRESENT, R8                                ;
                          57     0000'    CF  9E 0001E              MOVAB   P.AAJ, R7                                      ;
                          5E              34  C2 00023              SUBL2   #52, SP                                        ;
                          52        04    AC  D0 00026              MOVL    DATA_BUFFER, R2                                ; 0354
                          56        04    A2  9E 0002A              MOVAB   4(R2), R6                                      ;
                          55        0C    A2  9E 0002E              MOVAB   12(R2), R5                                     ;
                          54        14    A2  9E 00032              MOVAB   20(R2), R4                                     ;
                          53        1C    A2  9E 00036              MOVAB   28(R2), R3                                     ;
                          50        1C    AE  9E 0003A              MOVAB   JPI_LIST, $$ITMBLKPTR                          ; 0368
                          80 03190004     8F  D0 0003E              MOVL    #51970052, ($$ITMBLKPTR)+                      ;
                          80              6E  9E 00045              MOVAB   OURPID, ($$ITMBLKPTR)+                         ;
                                          80  7C 00048              CLRQ    ($$ITMBLKPTR)+                                 ;
                                          7E  7C 0004A              CLRQ    -(SP)                                          ; 0370
                                    34    AE  9F 0004C              PUSHAB  IOSB                                           ;
                                    28    AE  9F 0004F              PUSHAB  JPI_LIST                                       ;
                                          7E  7C 00052              CLRQ    -(SP)                                          ;
                                          7E  D4 00054              CLRL    -(SP)                                          ;
                 00000000G  00            07  FB 00056              CALLS   #7, SYS$GETJPIW                                ;
                                          57  DD 0005D              PUSHL   R7                                             ; 0376
```

```
                                                        I 4
SETPROCES                                    16-Sep-1984 00:45:54   VAX-11 Bliss-32 V4.0-742        Page  18
V04-000                                      14-Sep-1984 12:09:16   [CLIUTL.SRC]SETPROCES.B32;1          (6)

                             68          01  FB 0005F          CALLS    #1, CLI$PRESENT                ;
     62         01           01          50  F0 00062          INSV     R0, #1, #1, (R2)               ;
                  14  AE 020E0000        8F  D0 00067          MOVL     #34471936, DESC                ; 0381
                              18         AE  D4 0006F          CLRL     DESC+4                         ;
                              14         AE  9F 00072          PUSHAB   DESC                           ; 0382
                              10         A7  9F 00075          PUSHAB   P.AAL                          ;
                             6B          02  FB 00078          CALLS    #2, CLI$GET_VALUE              ;
                             38          50  E9 0007B          BLBC     R0, 4$                         ;
                             62          04  88 0007E          BISB2    #4, (R2)                       ; 0385
                              08         A2  9F 00081          PUSHAB   8(R2)                          ; 0386
                              1C         AE  DD 00084          PUSHL    DESC+4                         ; 0387
                             7E   1C     AE  3C 00087          MOVZWL   DESC, -(SP)                    ; 0386
               00000000G     00          03  FB 0008B          CALLS    #3, LIB$CVT_DTB                ;
                             05          50  E8 00092          BLBS     R0, 1$                         ;
                              20         A7  9F 00095          PUSHAB   P.AAN                          ; 0389
                                         0E  11 00098          BRB      3$                             ;
                     1F       08         A2  D1 0009A  1$:     CMPL     8(R2), #31                     ; 0392
                                         05  14 0009E          BGTR     2$                             ;
                              08         A2  D5 000A0          TSTL     8(R2)                          ; 0393
                                         11  18 000A3          BGEQ     4$                             ;
                              30         A7  9F 000A5  2$:     PUSHAB   P.AAP                          ; 0394
                              18         AE  9F 000A8  3$:     PUSHAB   DESC                           ;
                                         02  DD 000AB          PUSHL    #2                             ;
                       0077132A          8F  DD 000AD          PUSHL    #7803690                       ;
                             69          04  FB 000B3          CALLS    #4, LIB$SIGNAL                 ;
                              14         AE  9F 000B6  4$:     PUSHAB   DESC                           ; 0401
                              3C         A7  9F 000B9          PUSHAB   P.AAR                          ;
                             6B          02  FB 000BC          CALLS    #2, CLI$GET_VALUE              ;
                             26          50  E9 000BF          BLBC     R0, 6$                         ;
                             66          6E  D1 000C2          CMPL     OURPID, (R6)                   ; 0404
                                         0A  13 000C5          BEQL     5$                             ;
                              48         A7  9F 000C7          PUSHAB   P.AAT                          ; 0406
                                         01  DD 000CA          PUSHL    #1                             ; 0405
                                         5A  DD 000CC          PUSHL    R10                            ;
                             69          03  FB 000CE          CALLS    #3, LIB$SIGNAL                 ;
                             62          08  88 000D1  5$:     BISB2    #8, (R2)                       ; 0408
                             65   14     AE  3C 000D4          MOVZWL   DESC, (R5)                     ; 0409
                  04  A5      18         AE  D0 000D8          MOVL     DESC+4, 4(R5)                  ; 0410
                  14  AE 020E0000        8F  D0 000DD          MOVL     #34471936, DESC                ; 0411
                              18         AE  D4 000E5          CLRL     DESC+4                         ;
                              58         A7  9F 000E8  6$:     PUSHAB   P.AAV                          ; 0418
                             68          01  FB 000EB          CALLS    #1, CLI$PRESENT                ;
                             55          50  D0 000EE          MOVL     R0, STATUS                     ;
                             05          55  E9 000F1          BLBC     STATUS, 7$                     ; 0419
                             62          20  88 000F4          BISB2    #32, (R2)                      ; 0420
                                         0C  11 000F7          BRB      8$                             ;
               00000000G     8F          55  D1 000F9  7$:     CMPL     STATUS, #CLI$_NEGATED          ; 0421
                                         03  12 00100          BNEQ     8$                             ;
                             62          10  88 00102          BISB2    #16, (R2)                      ; 0422
                              68         A7  9F 00105  8$:     PUSHAB   P.AAX                          ; 0423
                             68          01  FB 00108          CALLS    #1, CLI$PRESENT                ;
                             03          50  E9 0010B          BLBC     R0, 9$                         ;
                             62          10  88 0010E          BISB2    #16, (R2)                      ; 0424
                              78         A7  9F 00111  9$:     PUSHAB   P.AAZ                          ; 0429
                             68          01  FB 00114          CALLS    #1, CLI$PRESENT                ;
                             55          50  D0 00117          MOVL     R0, STATUS                     ;
               00000000G     8F          55  D1 0011A          CMPL     STATUS, #CLI$_ABSENT           ; 0430
```

```
                                            1C   13 00121          BEQL     11$                                     ;
                                 66         6E   D1 00123          CMPL     OURPID, (R6)                            ; 0433
                                            0B   13 00126          BEQL     10$                                     ;
                                       0088 C7   9F 00128          PUSHAB   P.ABB                                   ; 0435
                                            01   DD 0012C          PUSHL    #1                                      ; 0434
                                            5A   DD 0012E          PUSHL    R10                                     ;
                                 69         03   FB 00130          CALLS    #3, LIB$SIGNAL                          ;
                                 62      40 8F   88 00133  10$:    BISB2    #64, (R2)                               ; 0437
                                 50         55   D2 00137          MCOML    STATUS, R0                              ; 0438
      62           01            07         50   F0 0013A          INSV     R0, #7, #1, (R2)                        ;
                                       00A0 C7   9F 0013F  11$:    PUSHAB   P.ABD                                   ; 0444
                                 68         01   FB 00143          CALLS    #1, CLI$PRESENT                         ;
                                 55         50   D0 00146          MOVL     R0, STATUS                              ;
                      00000000G  8F         55   D1 00149          CMPL     STATUS, #CLI$_ABSENT                    ; 0445
                                            1C   13 00150          BEQL     13$                                     ;
                                 66         6E   D1 00152          CMPL     OURPID, (R6)                            ; 0448
                                            0B   13 00155          BEQL     12$                                     ;
                                       00BC C7   9F 00157          PUSHAB   P.ABF                                   ; 0450
                                            01   DD 0015B          PUSHL    #1                                      ; 0449
                                            5A   DD 0015D          PUSHL    R10                                     ;
                                 69         03   FB 0015F          CALLS    #3, LIB$SIGNAL                          ;
                            01   A2         01   88 00162  12$:    BISB2    #1, 1(R2)                               ; 0452
                                 50         55   D2 00166          MCOML    STATUS, R0                              ; 0453
      62           01            09         50   F0 00169          INSV     R0, #9, #1, (R2)                        ;
                                       00D0 C7   9F 0016E  13$:    PUSHAB   P.ABH                                   ; 0459
                                 68         01   FB 00172          CALLS    #1, CLI$PRESENT                         ;
                                 03         50   E8 00175          BLBS     R0, 14$                                 ;
                                          0080   31 00178          BRW      18$                                     ;
                                 66         6E   D1 0017B  14$:    CMPL     OURPID, (R6)                            ; 0465
                                            0B   13 0017E          BEQL     15$                                     ;
                                       00E4 C7   9F 00180          PUSHAB   P.ABJ                                   ; 0467
                                            01   DD 00184          PUSHL    #1                                      ; 0466
                                            5A   DD 00186          PUSHL    R10                                     ;
                                 69         03   FB 00188          CALLS    #3, LIB$SIGNAL                          ;
                            01   A2         04   88 0018B  15$:    BISB2    #4, 1(R2)                               ; 0469
                                 50  00000000G  00   D0 0018F          MOVL     CTL$GQ_PROCPRIV, R0                     ; 0474
                            04   AE         50   D0 00196          MOVL     R0, NEWPRIV                             ;
                            0C   AE         50   D0 0019A          MOVL     R0, OLDPRIV                             ;
                                 50  00000000G  00   D0 0019E          MOVL     CTL$GQ_PROCPRIV+4, R0                   ; 0475
                            08   AE         50   D0 001A5          MOVL     R0, NEWPRIV+4                           ;
                            10   AE         50   D0 001A9          MOVL     R0, OLDPRIV+4                           ;
                                         14 AE   9F 001AD  16$:    PUSHAB   DESC                                    ; 0484
                                       00F8 C7   9F 001B0          PUSHAB   P.ABL                                   ;
                                 6B         02   FB 001B4          CALLS    #2, CLI$GET_VALUE                       ;
                                 27         50   E9 001B7          BLBC     R0, 17$                                 ;
                                         04 AE   9F 001BA          PUSHAB   NEWPRIV                                 ; 0487
                                         18 AE   9F 001BD          PUSHAB   DESC                                    ;
                      00000000G  00         02   FB 001C0          CALLS    #2, PRV$SETPRIV                         ;
                                 55         50   D0 001C7          MOVL     R0, STATUS                              ;
                                 E0         55   E8 001CA          BLBS     STATUS, 16$                             ;
                                       010C C7   9F 001CD          PUSHAB   P.ABN                                   ; 0489
                                         18 AE   9F 001D1          PUSHAB   DESC                                    ; 0488
                                            02   DD 001D4          PUSHL    #2                                      ;
                                   0077132A 8F   DD 001D6          PUSHL    #7803690                                ;
                                 69         04   FB 001DC          CALLS    #4, LIB$SIGNAL                          ;
                                            CC   11 001DF          BRB      16$                                     ; 0484
                   64       04   AE      0C AE   CB 001E1  17$:    BICL3    OLDPRIV, NEWPRIV, (R4)                  ; 0496
```

```
            04   A4       08   AE     10   AE  CB 001E7          BICL3   OLDPRIV+4, NEWPRIV+4, 4(R4)              ; 0497
                 63       0C   AE     04   AE  CB 001EE          BICL3   NEWPRIV, OLDPRIV, (R3)                   ; 0498
            04   A3       10   AE     08   AE  CB 001F4          BICL3   NEWPRIV+4, OLDPRIV+4, 4(R3)              ; 0499
                                    0118   C7  9F 001FB  18$:    PUSHAB  P.ABP                                    ; 0505
                               68          01  FB 001FF          CALLS   #1, CLI$PRESENT                          ;
                               55          50  D0 00202          MOVL    R0, STATUS                               ;
                    00000000G  8F          55  D1 00205          CMPL    STATUS, #CLI$_ABSENT                     ; 0506
                                           19  13 0020C          BEQL    20$                                      ;
                               66          6E  D1 0020E          CMPL    OURPID, (R6)                             ; 0509
                                           0B  13 00211          BEQL    19$                                      ;
                                    0124   C7  9F 00213          PUSHAB  P.ABR                                    ; 0511
                                           01  DD 00217          PUSHL   #1                                       ; 0510
                                           5A  DD 00219          PUSHL   R10                                      ;
                               69          03  FB 0021B          CALLS   #3, LIB$SIGNAL                           ;
                          01   A2          08  88 0021E  19$:    BISB2   #8, 1(R2)                                ; 0513
       62            01        0C          55  F0 00222          INSV    STATUS, #12, #1, (R2)                    ; 0514
                                               04 00227  20$:    RET                                              ; 0518

; Routine Size:  552 bytes,    Routine Base:  $CODE$ + 0181
```

```
  1 ROUTINE set_process (data_buffer) : NOVALUE =
  2 BEGIN
  2 !++
  2 !
  2 ! Set all the parameters specified, signalling any errors.
  2 !
  2 ! Inputs
  2 !       FLAGS will have bits set to indicate what is to change.
  2 !       PRIORITY will have the new priority.
  2 !       NEW_NAME will point to the new process name.
  2 !       PRIV will be the new privilege mask.
  2 !
  2 ! Outputs
  2 !       None
  2 !
  2 !--
  2 MAP
  2     data_buffer : REF VECTOR;
  2
  2 LOCAL
  2     status;
  2
  2 !
  2 ! Bind the data buffer to pleasant, simple names that humans can enjoy
  2 !
  2 bind_data;
  2
  2 !
  2 ! /PRIORITY = n
  2 !
  2 IF .flags[set$_priority]
  2 THEN
  3     BEGIN
  3     LOCAL
  3         want_priority;
  3
P 4     IF NOT (status = $SETPRI(PIDADR = pid,
  4                              PRI    = .priority))
  3     THEN SIGNAL(set$_writeerr, 1, %ASCID 'process priority',
  3             .status)
  3 !
  3 ! If the priority requested is greater than the base priority and the process
  3 ! does not have ALTPRI privilege then $SETPRI will only set the priority to the
  3 ! base.  If this is the case or the user requested a log then we need further
  3 ! information on the process to tell the user.  Since we cannot be sure if the
  3 ! wanted priority was set until after the $GETJPIW we must do it in all cases.
  3 !
  3     ELSE
  4     BEGIN
  4         LOCAL
  4             iosb : VECTOR[4,WORD],
  4             jpi_list : $ITMLST_DECL(ITEMS=2);
  4
  4         want_priority = .data_buffer[2];        ! Save the priority requested
  4                                                 ! Generic value because of BIND
  4 !
  4 ! Set up the JPI item list to get the new process priority.
```

```
;  582        0576  4 !
;  583      P 0577  4         $ITMLST_INIT(ITMLST = jpi_list,
;  584        0578  4                 (ITMCOD = jpi$_prib, BUFADR = priority));
;  585      P 0579  4         status = $GETJPIW(ITMLST = jpi_list,
;  586      P 0580  4                           PIDADR = pid,
;  587        0581  4                           IOSB   = iosb);
;  588        0582  4         IF .status
;  589        0583  4         THEN status = .iosb[0];
;  590        0584  4         IF NOT .status
;  591        0585  4         THEN SIGNAL(.status)
;  592        0586  4         ELSE
;  593        0587  4 !
;  594        0588  4 ! Display correct message
;  595        0589  4 !
;  596        0590  5         BEGIN
;  597        0591  5             IF .want_priority GTR .priority       ! If the desired priority was not
;  598        0592  5             THEN SIGNAL(set$_noprio)              ! set then ALTPRI not set
;  599        0593  5             ELSE
;  600        0594  5                 IF .flags[set$_log]               ! If logging requested
;  601        0595  5                 THEN SIGNAL(set$_prioset, 3, name_desc, .pid, .priority);
;  602        0596  4         END;
;  603        0597  3         END;
;  604        0598  2     END;
;  605        0599  2
;  606        0600  2 !
;  607        0601  2 ! /NAME = string
;  608        0602  2 !
;  609        0603  2 IF .flags[set$_name]
;  610        0604  2 THEN
;  611        0605  3     BEGIN
;  612        0606  4     IF NOT (status = $SETPRN(PRCNAM = new_name))
;  613        0607  3     THEN SIGNAL(set$_writeerr, 1,                        ! Signal if an error
;  614        0608  3                 %ASCID 'process name',
;  615        0609  3                 .status)
;  616        0610  3     ELSE IF .flags[set$_log]
;  617        0611  3     THEN SIGNAL(set$_nameset, 1, new_name);             ! or if /LOG
;  618        0612  2     END;
;  619        0613  2
;  620        0614  2 !
;  621        0615  2 ! /SUSPEND
;  622        0616  2 !
;  623        0617  2 IF .flags[set$_suspend]
;  624        0618  2 THEN
;  625        0619  3     BEGIN
;  626        0620  4     IF NOT (status = $SUSPND(PIDADR = pid))              ! If a problem,
;  627        0621  3     THEN SIGNAL(set$_notsuspnd, 2, name_desc, .pid,      ! signal it
;  628        0622  3                 .status)
;  629        0623  3     ELSE IF .flags[set$_log]                             ! If /LOG, signal it
;  630        0624  3     THEN SIGNAL(set$_suspnd, 2, name_desc, .pid);
;  631        0625  2     END;
;  632        0626  2
;  633        0627  2 !
;  634        0628  2 ! /NOSUSPEND or /RESUME
;  635        0629  2 !
;  636        0630  2 IF .flags[set$_resume]
;  637        0631  2 THEN
;  638        0632  3     BEGIN
;  639        0633 4      IF NOT (status = $RESUME(PIDADR = pid))
;  640        0634 3      THEN SIGNAL(set$_notresumed, 2, name_desc, .pid, .status)
;  641        0635 3      ELSE IF .flags[set$_log]
;  642        0636 3      THEN SIGNAL(set$_resumed, 2, name_desc, .pid);
;  643        0637 2      END;
;  644        0638 2
;  645        0639 2  !
;  646        0640 2  ! /[NO]SWAP
;  647        0641 2  !
;  648        0642 2  IF .flags[set$_swap]
;  649        0643 2  THEN
;  650        0644 3      BEGIN
;  651        0645 4      IF NOT (status = $SETSWM(SWPFLG = .flags[set$_swapval]))    ! If an error,
;  652        0646 3      THEN SIGNAL(set$_writeerr, 1, %ASCID 'swap mode',           ! signal it
;  653        0647 3              .status)
;  654        0648 3      ELSE IF .flags[set$_log]
;  655        0649 3      THEN SIGNAL(set$_modeset, 1,
;  656        0650 4                  (IF .flags[set$_swapval]
;  657        0651 4                   THEN %ASCID 'NOSWAP'
;  658        0652 3                   ELSE %ASCID 'SWAP'));
;  659        0653 2      END;
;  660        0654 2
;  661        0655 2  !
;  662        0656 2  ! /[NO]RESOURCE_WAIT
;  663        0657 2  !
;  664        0658 2  IF .flags[set$_wait]
;  665        0659 2  THEN
;  666        0660 3      BEGIN
;  667        0661 4      IF NOT (status = $SETRWM(WATFLG = .flags[set$_waitval]))
;  668        0662 3      THEN SIGNAL(set$_writeerr, 1,                         ! Signal if a problem
;  669        0663 3                  %ASCID 'resource wait mode',
;  670        0664 3                  .status)
;  671        0665 3      ELSE IF .flags[set$_log]                              ! Signal if /LOG
;  672        0666 3      THEN SIGNAL(set$_modeset, 1,
;  673        0667 4                  (IF .flags[set$_waitval]
;  674        0668 4                   THEN %ASCID 'NORESOURCE_WAIT'
;  675        0669 3                   ELSE %ASCID 'RESOURCE_WAIT'));
;  676        0670 2      END;
;  677        0671 2
;  678        0672 2  !
;  679        0673 2  ! /PRIVILEGES = list
;  680        0674 2  !
;  681        0675 2  IF .flags[set$_priv]
;  682        0676 2  THEN
;  683        0677 3      BEGIN
;  684        0678 3
;  685        0679 3  !
;  686        0680 3  ! Enable the new privileges.
;  687        0681 3  !
;  688        0682 3      IF .enab_priv[0] NEQ 0                        ! If anything to enable,
;  689        0683 3      OR .enab_priv[1] NEQ 0
;  690      P 0684 3      THEN status = $SETPRV(PRVADR = enab_priv,     ! do it and save the status,
;  691      P 0685 3                            PRMFLG = 1,
;  692        0686 4                            ENBFLG = 1)
;  693        0687 3      ELSE status = 1;                              ! otherwise set success.
;  694        0688 3
;  695        0689 3      IF .disab_priv[0] NEQ 0                       ! If anything to disable,
;   696           0690  3       OR .disab_priv[1] NEQ 0
;   697         P 0691  3       THEN $SETPRV(PRVADR = disab_priv,           ! do that as well.  Forget
;   698         P 0692  3                   PRMFLG = 1,                     ! the status, you can always
;   699           0693  3                   ENBFLG = 0);                    ! remove privilege.
;   700           0694  3
;   701           0695  3       IF NOT .status                              ! If it failed this time, then
;   702           0696  3       THEN SIGNAL(set$_notpriv, .status)          ! signal it
;   703           0697  3       ELSE
;   704           0698  4           BEGIN                                   ! NOTALLPRIV is success, so
;   705           0699  4           IF .status EQL ss$_notallpriv           ! signal it as an error
;   706           0700  4           THEN SIGNAL(ss$_notallpriv AND %X'FFFFFFFE')
;   707           0701  4           ELSE IF .flags[set$_log]                ! Signal a real success if
;   708           0702  4           THEN SIGNAL(set$_privset);              ! logging requested
;   709           0703  3           END;
;   710           0704  2       END;
;   711           0705  2
;   712           0706  2   !
;   713           0707  2   ! /[NO]DUMP
;   714           0708  2   !
;   715           0709  2   IF .flags[set$_dump]
;   716           0710  2   THEN
;   717           0711  2       IF .flags[set$_dumpval]
;   718           0712  2       THEN
;   719           0713  3           BEGIN
;   720         P 0714  3           $CMKRNL( ROUTIN = set_dump,
;   721           0715  3                   ARGLST = UPLIT(1,1));
;   722           0716  3           SIGNAL(set$_modeset, 1, %ASCID 'DUMP');
;   723           0717  3           END
;   724           0718  2       ELSE
;   725           0719  3           BEGIN
;   726         P 0720  3           $CMKRNL( ROUTIN = set_dump,
;   727           0721  3                   ARGLST = UPLIT(1,0));
;   728           0722  3           SIGNAL(set$_modeset, 1, %ASCID 'NODUMP');
;   729           0723  2           END;
;   730           0724  2
;   731           0725  2 RETURN;
;   732           0726  1 END;


                                                                     .PSECT  $PLIT$,NOWRT,NOEXE,2

74  69  72  6F  69  72  70  20  73  73  65  63  6F  72  70  001D4 P.ABU:  .ASCII  \process priority\
                                                        79  001E3
                                                  010E0010  001E4 P.ABT:  .LONG   17694736
                                                  00000000' 001E8         .ADDRESS P.ABU
            65  6D  61  6E  20  73  73  65  63  6F  72  70  001EC P.ABW:  .ASCII  \process name\
                                                  010E000C  001F8 P.ABV:  .LONG   17694732
                                                  00000000' 001FC         .ADDRESS P.ABW
            00  00  00  65  64  6F  6D  20  70  61  77  73  00200 P.ABY:  .ASCII  \swap mode\<0><0><0>
                                                  010E0009  0020C P.ABX:  .LONG   17694729
                                                  00000000' 00210         .ADDRESS P.ABY
                        00  00  50  41  57  53  4F  4E  00214 P.ACA:  .ASCII  \NOSWAP\<0><0>
                                                  010E0006  0021C P.ABZ:  .LONG   17694726
                                                  00000000' 00220         .ADDRESS P.ACA
                                        50  41  57  53  00224 P.ACC:  .ASCII  \SWAP\
                                                  010E0004  00228 P.ACB:  .LONG   17694724
```

```
                                                            00000000'  0022C             .ADDRESS P.ACC                                    ;
6D  20  74  69  61  77  20  65  63  72  75  6F  73  65  72  00230 P.ACE:  .ASCII  \resource wait mode\<0><0>                          ;
                                            00  00  65  64  6F  0023F                                                                    ;
                                                            010E0012   00244 P.ACD:  .LONG   17694738                                          ;
                                                            00000000'  00248         .ADDRESS P.ACE                                    ;
54  49  41  57  5F  45  43  52  55  4F  53  45  52  4F  4E  0024C P.ACG:  .ASCII  \NORESOURCE_WAIT\<0>                                ;
                                                            00  0025B                                                                    ;
                                                            010E000F   0025C P.ACF:  .LONG   17694735                                          ;
                                                            00000000'  00260         .ADDRESS P.ACG                                    ;
00  00  54  49  41  57  5F  45  43  52  55  4F  53  45  52  00264 P.ACI:  .ASCII  \RESOURCE_WAIT\<0><0><0>                            ;
                                                            00  00273                                                                    ;
                                                            010E000D   00274 P.ACH:  .LONG   17694733                                          ;
                                                            00000000'  00278         .ADDRESS P.ACI                                    ;
                                                  00000001  00000001   0027C P.ACJ:  .LONG   1, 1                                              ;
                                                        50  4D  55  44 00284 P.ACL:  .ASCII  \DUMP\                                            ;
                                                            010E0004   00288 P.ACK:  .LONG   17694724                                          ;
                                                            00000000'  0028C         .ADDRESS P.ACL                                    ;
                                                  00000000  00000001   00290 P.ACM:  .LONG   1, 0                                              ;
                                    00  00  50  4D  55  44  4F  4E     00298 P.ACO:  .ASCII  \NODUMP\<0><0>                                    ;
                                                            010E0006   002A0 P.ACN:  .LONG   17694726                                          ;
                                                            00000000'  002A4         .ADDRESS P.ACO                                    ;

                                                                             .EXTRN  SYS$SETPRI, SYS$SETPRN
                                                                             .EXTRN  SYS$SUSPND, SYS$RESUME
                                                                             .EXTRN  SYS$SETSWM, SYS$SETRWM
                                                                             .EXTRN  SYS$SETPRV, SYS$CMKRNL

                                                                             .PSECT  $CODE$,NOWRT,2

                                                         0FFC 00000 SET_PROCESS:
                                                                             .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11                    ; 0519
                                      5B 00000000G   8F  D0 00002            MOVL    #SET$_WRITEERR, R11                                     ;
                                      5A     0000'   CF  9E 00009            MOVAB   P.ABT, R10                                              ;
                                      59 00000000G   00  9E 0000E            MOVAB   LIB$SIGNAL, R9                                          ;
                                      5E             24  C2 00015            SUBL2   #36, SP                                                 ;
                                      52         04  AC  D0 00018            MOVL    DATA_BUFFER, R2                                         ; 0539
                                      53         04  A2  9E 0001C            MOVAB   4(R2), R3                                               ;
                                      57         14  A2  9E 00020            MOVAB   20(R2), R7                                              ;
                                      56         1C  A2  9E 00024            MOVAB   28(R2), R6                                              ;
                                      55         24  A2  9E 00028            MOVAB   36(R2), R5                                              ;
                      6C              62             02  E1 0002C            BBC     #2, (R2), 5$                                            ; 0549
                                                     7E  D4 00030            CLRL    -(SP)                                                   ; 0556
                                                 08  A2  DD 00032            PUSHL   8(R2)                                                   ;
                                                     7E  D4 00035            CLRL    -(SP)                                                   ;
                                                     53  DD 00037            PUSHL   R3                                                      ;
                          00000000G   00             04  FB 00039            CALLS   #4, SYS$SETPRI                                          ;
                                      54             50  D0 00040            MOVL    R0, STATUS                                              ;
                                      0D             54  E8 00043            BLBS    STATUS, 1$                                              ;
                                                     54  DD 00046            PUSHL   STATUS                                                  ; 0558
                                                     5A  DD 00048            PUSHL   R10                                                     ; 0557
                                                     01  DD 0004A            PUSHL   #1                                                      ;
                                                     5B  DD 0004C            PUSHL   R11                                                     ;
                                      69             04  FB 0004E            CALLS   #4, LIB$SIGNAL                                          ;
                                                     61  11 00051            BRB     7$                                                      ;
                                      58         08  A2  D0 00053 1$:        MOVL    8(R2), WANT_PRIORITY                                    ; 0572
                                      50             6E  9E 00057            MOVAB   JPI_LIST, $$ITMBLKPTR                                   ; 0578
                                      80 03090004    8F  D0 0005A            MOVL    #50921476, ($$ITMBLKPTR)+                               ;
```

```
                        80          08    A2  9E 00061         MOVAB   8(R2), ($$ITMBLKPTR)+              :
                                          80  7C 00065         CLRQ    ($$ITMBLKPTR)+                     :
                                          7E  7C 00067         CLRQ    -(SP)                              : 0581
                                    24    AE  9F 00069         PUSHAB  IOSB                               :
                                    0C    AE  9F 0006C         PUSHAB  JPI_LIST                           :
                                          7E  D4 0006F         CLRL    -(SP)                              :
                                          53  DD 00071         PUSHL   R3                                 :
                                          7E  D4 00073         CLRL    -(SP)                              :
              00000000G 00                07  FB 00075         CALLS   #7, SYS$GETJPIW                    :
                        54                50  D0 0007C         MOVL    R0, STATUS                         :
                        07                54  E9 0007F         BLBC    STATUS, 2$                         : 0582
                        54          1C    AE  3C 00082         MOVZWL  IOSB, STATUS                       : 0583
                        04                54  E8 00086         BLBS    STATUS, 3$                         : 0584
                                          54  DD 00089 2$:     PUSHL   STATUS                             : 0585
                                          0C  11 0008B         BRB     4$                                 :
                     08 A2                58  D1 0008D 3$:     CMPL    WANT_PRIORITY, 8(R2)               : 0591
                                          0B  15 00091         BLEQ    6$                                 :
                              00000000G   8F  DD 00093         PUSHL   #SET$_NOPRIO                       : 0592
                        69                01  FB 00099 4$:     CALLS   #1, LIB$SIGNAL                     :
                                          16  11 0009C 5$:     BRB     7$                                 :
         12             62                01  E1 0009E 6$:     BBC     #1, (R2), 7$                       : 0594
                                    08    A2  DD 000A2         PUSHL   8(R2)                              : 0595
                                          63  DD 000A5         PUSHL   (R3)                               :
                                          55  DD 000A7         PUSHL   R5                                 :
                                          03  DD 000A9         PUSHL   #3                                 :
                              00000000G   8F  DD 000AB         PUSHL   #SET$_PRIOSET                      :
                        69                05  FB 000B1         CALLS   #5, LIB$SIGNAL                     :
         30             62                03  E1 000B4 7$:     BBC     #3, (R2), 9$                       : 0603
                                    0C    A2  9F 000B8         PUSHAB  12(R2)                             : 0606
              00000000G 00                01  FB 000BB         CALLS   #1, SYS$SETPRN                     :
                        54                50  D0 000C2         MOVL    R0, STATUS                         :
                        0E                54  E8 000C5         BLBS    STATUS, 8$                         :
                                          54  DD 000C8         PUSHL   STATUS                             : 0609
                                    14    AA  9F 000CA         PUSHAB  P.ABV                              : 0607
                                          01  DD 000CD         PUSHL   #1                                 :
                                          5B  DD 000CF         PUSHL   R11                                :
                        69                04  FB 000D1         CALLS   #4, LIB$SIGNAL                     :
                                          12  11 000D4         BRB     9$                                 :
         0E             62                01  E1 000D6 8$:     BBC     #1, (R2), 9$                       : 0610
                                    0C    A2  9F 000DA         PUSHAB  12(R2)                             : 0611
                                          01  DD 000DD         PUSHL   #1                                 :
                              00000000G   8F  DD 000DF         PUSHL   #SET$_NAMESET                      :
                        69                03  FB 000E5         CALLS   #3, LIB$SIGNAL                     :
         37             62                05  E1 000E8 9$:     BBC     #5, (R2), 11$                      : 0617
                                          7E  D4 000EC         CLRL    -(SP)                              : 0620
                                          53  DD 000EE         PUSHL   R3                                 :
              00000000G 00                02  FB 000F0         CALLS   #2, SYS$SUSPND                     :
                        54                50  D0 000F7         MOVL    R0, STATUS                         :
                        13                54  E8 000FA         BLBS    STATUS, 10$                        :
                                          54  DD 000FD         PUSHL   STATUS                             : 0622
                                          63  DD 000FF         PUSHL   (R3)                               : 0621
                                          55  DD 00101         PUSHL   R5                                 :
                                          02  DD 00103         PUSHL   #2                                 :
                              00000000G   8F  DD 00105         PUSHL   #SET$_NOTSUSPND                    :
                        69                05  FB 0010B         CALLS   #5, LIB$SIGNAL                     :
                                          13  11 0010E         BRB     11$                                :
         0F             62                01  E1 00110 10$:    BBC     #1, (R2), 11$                      : 0623
```

```
                                63 DD 00114       PUSHL   (R3)                          ; 0624
                                55 DD 00116       PUSHL   R5                            ;
                                02 DD 00118       PUSHL   #2                            ;
                      00000000G 8F DD 0011A       PUSHL   #SET$_SUSPND                  ;
                  69            04 FB 00120       CALLS   #4, LIB$SIGNAL                ;
    37            62            04 E1 00123 11$:  BBC     #4, (R2), 13$                 ; 0630
                                7E D4 00127       CLRL    -(SP)                         ; 0633
                                53 DD 00129       PUSHL   R3                            ;
        00000000G 00            02 FB 0012B       CALLS   #2, SYS$RESUME                ;
                  54            50 D0 00132       MOVL    R0, STATUS                    ;
                  13            54 E8 00135       BLBS    STATUS, 12$                   ;
                                54 DD 00138       PUSHL   STATUS                        ; 0634
                                63 DD 0013A       PUSHL   (R3)                          ;
                                55 DD 0013C       PUSHL   R5                            ;
                                02 DD 0013E       PUSHL   #2                            ;
                      00000000G 8F DD 00140       PUSHL   #SET$_NOTRESUMED              ;
                  69            05 FB 00146       CALLS   #5, LIB$SIGNAL                ;
                                13 11 00149       BRB     13$                           ;
    0F            62            01 E1 0014B 12$:  BBC     #1, (R2), 13$                 ; 0635
                                63 DD 0014F       PUSHL   (R3)                          ; 0636
                                55 DD 00151       PUSHL   R5                            ;
                                02 DD 00153       PUSHL   #2                            ;
                      00000000G 8F DD 00155       PUSHL   #SET$_RESUMED                 ;
                  69            04 FB 0015B       CALLS   #4, LIB$SIGNAL                ;
    3F            62            06 E1 0015E 13$:  BBC     #6, (R2), 17$                 ; 0642
7E  62            01            07 EF 00162       EXTZV   #7, #1, (R2), -(SP)           ; 0645
        00000000G 00            01 FB 00167       CALLS   #1, SYS$SETSWM                ;
                  54            50 D0 0016E       MOVL    R0, STATUS                    ;
                  0E            54 E8 00171       BLBS    STATUS, 14$                   ;
                                54 DD 00174       PUSHL   STATUS                        ; 0647
                             28 AA 9F 00176       PUSHAB  P.ABX                         ; 0646
                                01 DD 00179       PUSHL   #1                            ;
                                5B DD 0017B       PUSHL   R11                           ;
                  69            04 FB 0017D       CALLS   #4, LIB$SIGNAL                ;
                                1F 11 00180       BRB     17$                           ;
    1B            62            01 E1 00182 14$:  BBC     #1, (R2), 17$                 ; 0648
                                62 95 00186       TSTB    (R2)                          ; 0650
                                06 18 00188       BGEQ    15$                           ;
                  50         38 AA 9E 0018A       MOVAB   P.ABZ, R0                     ; 0651
                                04 11 0018E       BRB     16$                           ;
                  50         44 AA 9E 00190 15$:  MOVAB   P.ACB, R0                     ; 0652
                                50 DD 00194 16$:  PUSHL   R0                            ;
                                01 DD 00196       PUSHL   #1                            ; 0649
                      00000000G 8F DD 00198       PUSHL   #SET$_MODESET                 ;
                  69            03 FB 0019E       CALLS   #3, LIB$SIGNAL                ;
                  40         01 A2 E9 001A1 17$:  BLBC    1(R2), 21$                    ; 0658
7E                62            09 EF 001A5       EXTZV   #9, #1, (R2), -(SP)           ; 0661
        00000000G 00            01 FB 001AA       CALLS   #1, SYS$SETRWM                ;
                  54            50 D0 001B1       MOVL    R0, STATUS                    ;
                  0E            54 E8 001B4       BLBS    STATUS, 18$                   ;
                                54 DD 001B7       PUSHL   STATUS                        ; 0664
                             60 AA 9F 001B9       PUSHAB  P.ACD                         ; 0662
                                01 DD 001BC       PUSHL   #1                            ;
                                5B DD 001BE       PUSHL   R11                           ;
                  69            04 FB 001C0       CALLS   #4, LIB$SIGNAL                ;
                                20 11 001C3       BRB     21$                           ;
    1C            62            01 E1 001C5 18$:  BBC     #1, (R2), 21$                 ; 0665
```

```
             06              62               09  E1 001C9          BBC      #9, (R2), 19$                                    ; 0667
                             50        78     AA  9E 001CD          MOVAB    P.ACF, R0                                        ; 0668
                                              05  11 001D1          BRB      20$
                             50      0090     CA  9E 001D3 19$:     MOVAB    P.ACM, R0                                        ; 0669
                                              50  DD 001D8 20$:     PUSHL    R0
                                              01  DD 001DA          PUSHL    #1                                               ; 0666
                                  00000000G   8F  DD 001DC          PUSHL    #SET$_MODESET
                             69               03  FB 001E2          CALLS    #3, LIB$SIGNAL
             63              62               0A  E1 001E5 21$:     BBC      #10, (R2), 30$                                   ; 0675
                                              67  D5 001E9          TSTL     (R7)                                             ; 0682
                                              05  12 001EB          BNEQ     22$
                                       04     A7  D5 001ED          TSTL     4(R7)                                            ; 0683
                                              13  13 001F0          BEQL     23$
                             7E               01  7D 001F2 22$:     MOVQ     #1, -(SP)                                        ; 0686
                                              57  DD 001F5          PUSHL    R7
                                              01  DD 001F7          PUSHL    #1
                 00000000G   00               04  FB 001F9          CALLS    #4, SYS$SETPRV
                             54               50  D0 00200          MOVL     R0, STATUS
                                              03  11 00203          BRB      24$                                              ; 0684
                             54               01  D0 00205 23$:     MOVL     #1, STATUS                                       ; 0687
                                              66  D5 00208 24$:     TSTL     (R6)                                             ; 0689
                                              05  12 0020A          BNEQ     25$
                                       04     A6  D5 0020C          TSTL     4(R6)                                            ; 0690
                                              0E  13 0020F          BEQL     26$
                             7E               01  7D 00211 25$:     MOVQ     #1, -(SP)                                        ; 0693
                                              56  DD 00214          PUSHL    R6
                                              7E  D4 00216          CLRL     -(SP)
                 00000000G   00               04  FB 00218          CALLS    #4, SYS$SETPRV
                             0D               54  E8 0021F 26$:     BLBS     STATUS, 27$                                      ; 0695
                                              54  DD 00222          PUSHL    STATUS                                           ; 0696
                                  00000000G   8F  DD 00224          PUSHL    #SET$_NOTPRIV
                             69               02  FB 0022A          CALLS    #2, LIB$SIGNAL
                                              1D  11 0022D          BRB      30$
                 00000681    8F               54  D1 0022F 27$:     CMPL     STATUS, #1665                                    ; 0699
                                              07  12 00236          BNEQ     28$
                             7E      0680     8F  3C 00238          MOVZWL   #1664, -(SP)                                     ; 0700
                                              0A  11 0023D          BRB      29$
             09              62               01  E1 0023F 28$:     BBC      #1, (R2), 30$                                    ; 0701
                                  00000000G   8F  DD 00243          PUSHL    #SET$_PRIVSET                                    ; 0702
                             69               01  FB 00249 29$:     CALLS    #1, LIB$SIGNAL
             37              62               0B  E1 0024C 30$:     BBC      #11, (R2), 33$                                   ; 0709
             15              62               0C  E1 00250          BBC      #12, (R2), 31$                                   ; 0711
                                     0098     CA  9F 00254          PUSHAB   P.ACJ                                            ; 0715
                                     0000V    CF  9F 00258          PUSHAB   SET_DUMP
                 00000000G   00               02  FB 0025C          CALLS    #2, SYS$CMKRNL
                                     00A4     CA  9F 00263          PUSHAB   P.ACK                                            ; 0716
                                              13  11 00267          BRB      32$
                                     00AC     CA  9F 00269 31$:     PUSHAB   P.ACM                                            ; 0721
                                     0000V    CF  9F 0026D          PUSHAB   SET_DUMP
                 00000000G   00               02  FB 00271          CALLS    #2, SYS$CMKRNL
                                     00BC     CA  9F 00278          PUSHAB   P.ACN                                            ; 0722
                                              01  DD 0027C 32$:     PUSHL    #1
                                  00000000G   8F  DD 0027E          PUSHL    #SET$_MODESET
                             69               03  FB 00284          CALLS    #3, LIB$SIGNAL
                                                  04 00287 33$:     RET                                                       ; 0726

; Routine Size:  648 bytes,    Routine Base:  $CODE$ + 03A9
```

```
;  734          0727  1 ROUTINE set_dump (mode) : NOVALUE =
;  735          0728  2 BEGIN
;  736          0729  2 !++
;  737          0730  2 ! Functional description
;  738          0731  2 !
;  739          0732  2 !       This routine sets the dump mode.  It can only affect the current
;  740          0733  2 !       process.
;  741          0734  2 !
;  742          0735  2 ! Inputs
;  743          0736  2 !       mode - 1 or 0 for mode on or off
;  744          0737  2 !
;  745          0738  2 ! Outputs
;  746          0739  2 !       None
;  747          0740  2 !
;  748          0741  2 !----
;  749          0742  2
;  750          0743  2 ctl$gl_phd[phd$v_imgdmp] = .mode;
;  751          0744  2
;  752          0745  2 return;
;  753          0746  2
;  754          0747  1 END;



                                                0000 00000 SET_DUMP:
                                                               .WORD   Save nothing                              ; 0727
                                  50 00000000G  00  D0 00002   MOVL    CTL$GL_PHD, R0                            ; 0743
      36   A0           01        05        04  AC  F0 00009   INSV    MODE, #5, #1, 54(R0)                      ;
                                                    04 00010   RET                                               ; 0747

; Routine Size:  17 bytes,     Routine Base:  $CODE$ + 0631


;  755          0748  1
;  757          0749  1 END
;  758          0750  0 ELUDOM



                                                                  .EXTRN  LIB$SIGNAL

;                                 PSECT SUMMARY
;
;       Name                          Bytes                      Attributes
;
;  $PLIT$                              680  NOVEC,NOWRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $CODE$                             1602  NOVEC,NOWRT,  RD ,  EXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)



;                            Library Statistics
;
;                                               -------- Symbols --------       Pages        Processing
;       File                                    Total    Loaded   Percent       Mapped       Time
;
;  _$255$DUA28:[SYSLIB]LIB.L32;1                18619        43         0       1000          00:01.8



; Information:  1
; Warnings:     0
; Errors:       0



;                                 COMMAND QUALIFIERS

;       BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LIS$:SETPROCES/OBJ=OBJ$:SETPROCES MSRC$:SETPROCES/UPDATE=(ENH$:SETPROCES)

; Size:          1602 code + 680 data bytes
; Run Time:         00:31.1
; Elapsed Time:     01:46.8
; Lines/CPU Min:     1447
; Lexemes/CPU-Min: 22162
; Memory Used:  228 pages
; Compilation Complete
```

0054 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

SETPROCES
LIS

SETSHOBRO
LIS

SETVOLUME
LIS

SETPWD
LIS

SETTERM
LIS

SETQUEUE
LIS

SETTIME
LIS